A SYSTEM AND METHOD FOR SERVER-BASED PREDICTIVE CACHING OF BACK-END SYSTEM DATA

BACKGROUND OF THE INVENTION

Field of Invention

[1] The present invention relates generally to client/server networks and, more particularly, to a system and method for accelerating client access of remote information items by predicting user actions.

Description of the Background

[2] Client-based software programs called "web accelerators" address the problem of file system latency by pre-fetching web pages from a server device in advance of a request from an associated client. Typically, web accelerators come in the form of a browser plug-in that can recognize the links present on the displayed web page and start to download the information they represent, in advance of a user request, into a storage area of the client computer called a memory cache. By the time the client user is finished viewing the displayed page and clicks on the next link, the information associated with that next link is already available in the client user's memory cache. Other software programs allow the web page developer to assign a probability to each link, or otherwise indicate a preference, which corresponds to the likelihood that the client user will choose that link next. Those links that the user is most likely to choose are assigned the highest probability while rarely accessed links are assigned little or no weight. Despite these innovations, users still experience significant delays when an information request causes the server to access the information from back-end systems. Back-end systems, as used herein, refer to remote systems such as data warehouse repositories or legacy systems.

[3] A system and method is needed for reducing delays in accessing information from backend systems. The system should also have sufficient flexibility to selectively retrieve information items from the backend system. The system should also be capable of accelerating client access to backend systems without unnecessarily consuming bandwidth between the client and server devices. Finally, the system needs to be simple to implement on a wide-scale basis without requiring upgrades to individual browsers.

SUMMARY OF THE INVENTION

[4] The present invention is directed to a system and method for accelerating client access of remote information items by predicting user actions. According to one embodiment, the system includes a remote data storage device for storing the information items, wherein the information items are stored in the form of pages, and wherein the pages contain a plurality of links to other information items; a client device having a user interface program thereon, for allowing a user to interface with the network and request the information items; a server device, in communication with the client device and in communication with the remote storage device, for handling information requests from multiple clients and for storing information retrieved from the data storage devices locally in a server cache memory; a data collection module for collecting and storing successive user actions; and a probability module in communication with the data collection module for calculating a probability for the links, and for comparing the probability to a predetermined threshold value, and for retrieving the information items associated with the links from the remote data storage devices and storing the information items in the server cache memory in advance of a user request for the selected information items.

[5] The system and method of the present invention may be used, for example, to accelerate client access of remote information items by predicting user actions. For example, the

present invention may be used in the provision of electronic commerce services in which certain information items reside on a remote storage device, such as a legacy system. These and other benefits of the present invention will be apparent from the detailed description below.

DESCRIPTION OF THE FIGURES

[6] For the present invention to be understood clearly and readily practiced, the present invention will be described in conjunction with the following figures, wherein:

[7] FIG. 1 is a block diagram illustrating an implementation of a high level multi-layer architecture that provides predictive file caching to address the problem of file system latency according to one embodiment of the present invention;

[8] FIG. 2 is schematic diagram of a server device according to one embodiment of the present invention;

[9] FIG. 3 is a relational diagram illustrating all possible user-selected navigational events in a web site according to one embodiment of the present invention; and

[10] FIG. 4 is a flow diagram illustrating the general method for accelerating client access of remote information resources by predicting user actions according to one embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[11] It is to be understood that the figures and descriptions of the present invention have been simplified to illustrate elements that are relevant for a clear understanding of the present invention while eliminating, for purposes of clarity, other elements. For example, certain network architecture details and interoperability applications are not described herein. Those of ordinary skill in the art will recognize, however, that these and other elements may be desirable

in a typical client/server network functionally connected to a backend system. A discussion of such elements is not provided because such elements are well known in the art and because they do not facilitate ax better understanding of the present invention.

[12] FIG. 1 shows an exemplary implementation of a high level multi-layer architecture that provides predictive file caching to address the problem of file system latency in a client/server network functionally connected to a backend system. In particular, FIG. 1 shows a client/server system 10 that includes at least one client device 16, a wide or local area network 17, at least one application server device 18, an enterprise application integration (EAI) application 20, a middleware application 22, a legacy system 24, and legacy data 26. According to one embodiment, the client and server devices 16, 18 are standard computer systems. Those skilled in the art will appreciate that the hardware and software interfaces between elements shown in FIG. 1 are conventional and do not form part of the invention.

[13] A standard computer system includes a processor with standard input/output interfaces, a user interface such as a conventional keyboard and video display for accessing a central processing unit, high speed random access memory (RAM), and a physical storage device such as a fixed disk drive, upon which the operating system, application programs, and data files are stored. Those skilled in the art will appreciate that this is a simplified example of a computer and that other hardware devices and input/output connections may be present.

[14] The client device 16 stores a number of computer programs, including a "browser." As known in the art, a browser is used to communicate with the remote server device 18 and to visually present the information received from such a device. The server device 18, shown in greater detail in FIG. 2, includes a processor 28 connected to RAM 30 and a physical storage device 32. The RAM 30 includes a file system 34 that stores a computer program 35 to

cache files in a portion of RAM 30 shown as cache 38. It will be appreciated, however, that the invention can be implemented at any level, ranging from hardware to application software.

[15] The enterprise application integration (EAI) and middleware applications 20, 22 are system software applications that help programs and databases work together on diverse systems. According to the embodiment illustrated in FIG. 1, the EAI and middleware applications 20, 22 integrate other applications executed on the server 18 with the data 12 stored on legacy systems 24. Those skilled in the art will appreciate that the software layers 20, 22, 24 shown in FIG. 1 are conventional and do not form part of the invention. Those skilled in the art will also appreciate that these software layers 20, 22, 24 are shown only to illustrate a cause for file system latency.

[16] The file system 34 is supported by a data collection module 36 for tracking sequences of navigational events and a probability calculator module 40 for predicting likely future events, based on tracked sequences, and pre-fetching data for future events from remote storage 26, or disk 32, and caching the data in cache 38. The predictive caching method, therefore, can be described in terms of two principal components: (i) a data collection module 36 for tracking sequences of previous events and (ii) a probability calculator 40 for using the tracked sequences to determine likely future events and pre-fetching data for the likely future events.

[17] The data collection module 36 collects and stores, for each web link on a given web site, the sequence of web objects selected by all users. FIG. 3 shows an exemplary relational diagram 50 that illustrates the possible user-selected navigational events with respect to a particular link 52 to a web object, which resides on a web page 54 (not shown). The web

page 54, in turn, is one of a plurality of web pages that form a web site 56 (also not shown). These designations will become useful in the description of the embodiments as set forth below.

[18] By way of example, the link 52 is a banner advertisement in which a voice and data communications provider offers a discount on an additional telephone line. The diagram 50 maps the choices 57, 58, 60 presented to a site visitor (or "user") with respect to the link 52. The user may choose to buy the second phone line by clicking on the link 52 (choice 57). Alternatively, the user may choose a link to a related section of the web site 56 (choice 58) or a link to an unrelated part of the web site 56 (choice 60). The data collection module 36 collects and stores these user actions for each link in the web site, thereby creating a comprehensive event history with respect to each link. According to another embodiment, the collection module 36 may collect and store the navigation information on a user-specific basis. For example, if a user logs onto the web site 56 using a conventional authentication procedure, the collection module 36 can attribute all subsequent navigational events to that user. When the user returns to the web site 56, the probability module 40 can make custom predictions based on the user's navigational choices during prior visits. According to another embodiment, which persons of ordinary skill in the art will appreciate, the data collection activity can take place on an ongoing basis so that the database 32 maintained by data collection module 36 is updated with each successive user activity.

[19] The probability module 40 determines the likelihood that future users will select a given web object using the data collected and stored by the data collection module 36. For example, assume the data collection module 36 contains navigation information for 10 unauthenticated users, 3 of which chose a link to an area of the web site 56 unrelated to the link 52 (choice 60). The probability module 40 will determine that the next user has a 0.3 probability

of choosing an unrelated link (choice 60). According to another embodiment, if a subsequent user chooses an unrelated link (choice 60) the data collection module 36 updates its event history and the probability module 40 updates the probability calculation. According to this example, the probability module 40 will update the probability that the next user will choose an unrelated site to approximately 0.36 (4 of 11 users). The probability module will also update probabilities assigned to the other two choices 56, 58 in the same manner so that the sum of the three probabilities equals one.

[20] The probability module 40 also compares the probability to a pre-determined threshold value to determine whether the linked information item should be downloaded. For example, the probability module 40 may include a rules engine that contains a plurality of business rules for establishing the threshold. The business rules may consider, for example, the level of risk of retrieving data that may not be used. The level of risk may be attributed to, for example, the hardware cost of cache memory. If the probability exceeds the pre-determined threshold, the cache 38 is augmented by pre-fetching data that are likely to be accessed. Otherwise, no data will be pre-fetched.

[21] FIG. 4 is a flow chart illustrating the general method for accelerating client access of remote information items by predicting user actions according to one embodiment of the present invention. FIG. 4 is entered at step 100, and in step 102, the present invention displays the web page 54 on the client device 16. In step 104, the probability associated with the link 52 on page 54 is read in a conventional manner and, in step 106, compared to the predetermined threshold value. If the probability of the user clicking on the link 52 (choice 57) is less than the threshold value, the present invention proceeds to an end in step 107.

[22] If the probability of the user clicking on the link 52 (choice 57) is greater than the threshold value, then in step 108 the program 35 of the present invention begins to download the predicted item from remote storage 26. As explained above, the program 35 stores the predicted item in the cache 38. If the link selection determination in step 110 is incorrect, then in step 112 the present invention aborts the download of the predicted item and proceed to the end in step 107. Next, in step 114, the present invention commences a download of the selected item. If the prediction in step 110 is correct, the present invention continues to download the item in a conventional manner from the remote system 24 to the cache 38 in the application server 18.

[23] It should be understood that the invention is not limited by the foregoing description, but embraces all such alterations, modifications, and variations in accordance with the spirit and scope of the appended claims.